MELTR: Meta Loss Transformer for Learning to Fine-tune Video Foundation Models (Supplementary Materials)

Dohwan Ko[1*] Joonmyung Choi[1*] Hyeong Kyu Choi[1]
Kyoung-Woon On[2] Byungseok Roh[2] Hyunwoo J. Kim[1†]

[1]Department of Computer Science and Engineering, Korea University [2]Kakao Brain
{ikodoh, pizard, imhgchoi, hyunwoojkim}@korea.ac.kr
{kloud.ohn, peter.roh}@kakaobrain.com

A. Implementation Details

A.1. Backbone Foundation Models

UniVL [1]. Our implementation is based on the official code of UniVL [2] pretrained on the HowTo100M dataset [3]. As in the main paper, we use eight auxiliary loss functions: $\mathcal{L}_{\text{Joint}}$, $\mathcal{L}_{\text{M-Joint}}$, $\mathcal{L}_{\text{Align}}$, $\mathcal{L}_{\text{M-Align}}$, $\mathcal{L}_{\text{CMLM}}$, $\mathcal{L}_{\text{CMFM}}$, $\mathcal{L}_{\text{Decoder}}$, and $\mathcal{L}_{\text{M-Decoder}}$. For the primary losses for text-to-video retrieval and video captioning tasks are $\mathcal{L}_{\text{Align}}$ and $\mathcal{L}_{\text{Decoder}}$, respectively. $\mathcal{L}_{\text{Align}}$ is also used as the primary loss function for multi-modal sentiment analysis.

Violet [4]. We implement MELTR based on the official Violet github [5] pretrained on the YT-Temporal 180M [6], WebVid [7], and CC3M [8]. For text-to-video retrieval, we adopt three auxiliary losses: video-text matching loss, masked text modeling loss, and masked visual-token modeling. We use the former one as the primary task loss. We use additional classification loss for video question answering.

All-in-one [9]. Our implementation for All-in-one is based on [10] and it is pretrained on WebVid [7], YT-Temporal 180M [6], HowTo100M [3], CC3M [8], CC12M [11], COCO [12], VisualGenome [13], and SBU [14]. When conducting text-to-video retrieval task, video-text matching loss and masked language modeling loss are adopted and the former one is used as the primary loss.

A.2. Evaluation metrics

For the video retrieval task, we report the standard retrieval metrics, Recall at K (R@K) metric (K=1,5,10) and Median Rank (MedR). Accuracy metric is reported for video question answering task which includes both multi-choice and open-ended questions. As for video captioning, BLEU [15], METEOR [16], ROUGE-L [17], and CIDEr [18] are reported.

A.3. MELTR Details.

We use the Adam [19] optimizer with an initial learning rate α = 3e-5 and β = 1e-4 with a linear learning rate decay strategy. For MELTR, we use one transformer encoder layer with 8 attention heads and 512 hidden dimensions. We trained 40, 20, and 20 epochs on the text-to-video retrieval, video question answering, and video captioning tasks with 8 $\times$ Tesla A100 GPUs, respectively. We search γ in $\{0.1, 0.3, 0.5\}$ for the regularization term and use $K = 3$ in Eq. (13) of the main paper.

B. Dataset Details

YouCook2. YouCook2 [20] consists of 2k videos, which cover 89 types of recipes. Each video contains multiple video clips accompanied by text descriptions. The train dataset contains 1,261 samples, and the test set contains 439 samples, respectively.

MSRVTT. The original MSRVTT-full [21] dataset, used on video captioning task, contains 6,513 train, 497 validation, and 2,990 test samples. However, we have observed a wide range of dataset split variations throughout research on text-to-video retrieval. One split variant randomly samples 1,000 clip-text pairs from the test set for evaluation and uses the rest of the 9,000 samples as train data [22], which is commonly denoted as the 1kA split. On the other hand, the 1kB split uses the identical 1,000 test split of 1kA for the test, whereas the train set is a subset of 1kA's containing 6,656 samples [23]. Another commonly used data split also uses the identical 1,000 test set, while adopting both the train and validation set from the standard MSRVTT for training. We evaluated our method on two split protocols most prominently observed in the literature, 1kA, and 7k.

*Equal contribution.
†Corresponding author.

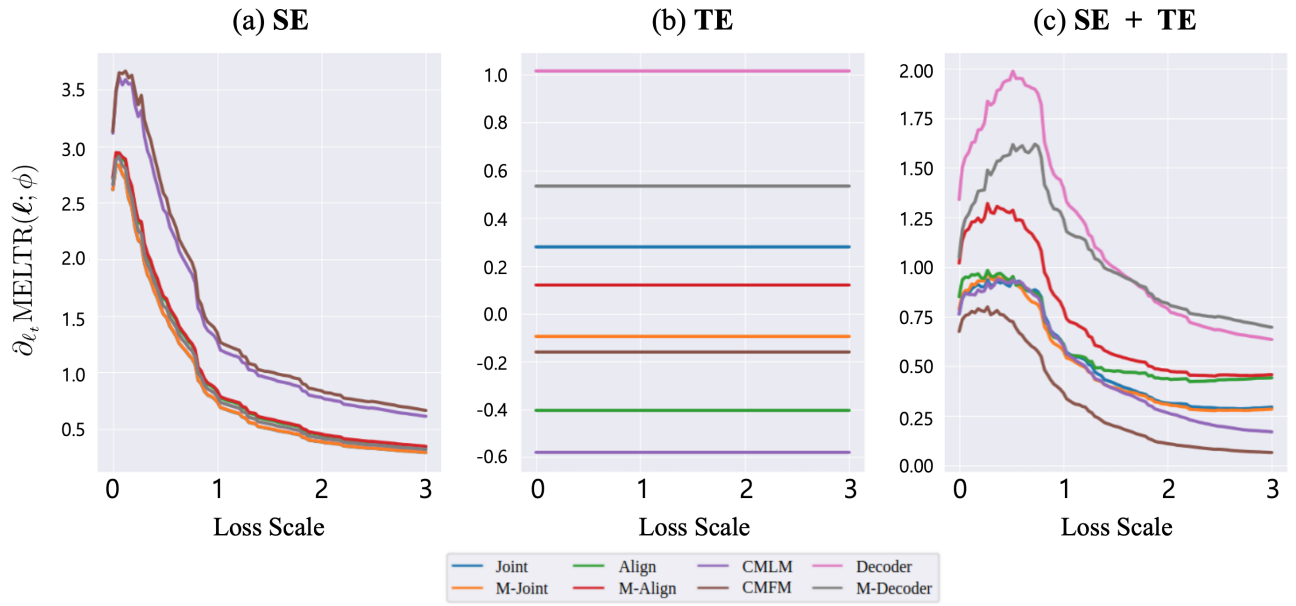

Figure 1. **Gradient by embedding type.** The gradient of MELTR output with respect to each task loss is plotted for different input embedding types. (a) Gradient values are generally similar across tasks, and only those with distinct loss scales are distinguished. (b) Gradients are different across tasks, but stay constant along loss scale, as loss scale information is not provided. (c) MELTR learned to effectively consider both loss scale and task information.

For convenience, we denote the former as MSRVTT-9k and the latter as MSRVTT-7k.

TGIF-QA. TGIF-QA [24] contains 165k QA pairs of animated GIFs. The dataset provides three different subtasks: TGIF-Action, TGIF-Transition and TGIF-Frame. TGIF-Action is to identify repeated actions, TGIF-Transition is to identify the transition between states, and TGIF-Frame is to answer questions given a GIF frame. TGIF-Action and TGIF-Transition are conducted under the multi-choice question answering setting, predicting the best answer given five options. TGIF-Frame is experimented as the open-ended question answering with 1,540 most frequent answer candidates.

MSVD-QA. MSVD-QA [25] contains 47k open-ended questions on 2k videos, derived from the original MSVD dataset [26]. We construct the answer set with 1,000 most frequently appeared answers.

CMU-MOSI. For the multi-modal sentiment analysis task, we adopt the CMU-MOSI dataset [27] which consists of 2,199 opinion video clips annotated with sentiment intensity values from -3 to 3.

C. Effectiveness of the Regularization Term

We proposed the regularization term $\mathcal{L}^{\text{reg}}$ in Section 4.1 of the main paper. Eq. (6) of the main paper encourages the learned loss MELTR$(\ell;\phi)$ to stay within a reasonable range to avoid meta-overfitting. Table 1 shows the ablation study for $\mathcal{L}^{\text{reg}}$ by adjusting the regularization strength γ on the text-to-video retrieval of MSRVTT-7k. Without the regularization term, *i.e.*, $\gamma = 0$, it shows the performance of 27.6% on R@1 metric. The performance improves at $\gamma = 1$ or $\gamma = 10$ by a margin of 1% than without $\mathcal{L}^{\text{reg}}$.

Table 1. **Regularization strength.**

γ	0	0.001	0.01	0.1	1.0	10	100
R@1	27.6	27.8	28.1	28.4	**28.6**	**28.6**	28.5

D. Effectiveness of transformer architecture

In this section, we conduct an ablation study for the architecture type of MELTR on the text-to-video retrieval on MSRVTT by replacing the transformer with a linear layer. Table 2 demonstrates that the transformer architecture improves by margin of 1% than the linear layer by taking advantage of the self-attention layer. Furthermore, we use both the scale embedding and task embedding (**SE** + **TE**) as the input of MELTR. Only with **SE**, MELTR cannot consider task information and hence the performance decreases. However, only with **TE**, MELTR cannot be trained since the input losses are not passed to MELTR, *i.e.*, $\nabla_w \mathcal{L}^{\text{aux}}$ is always zero.

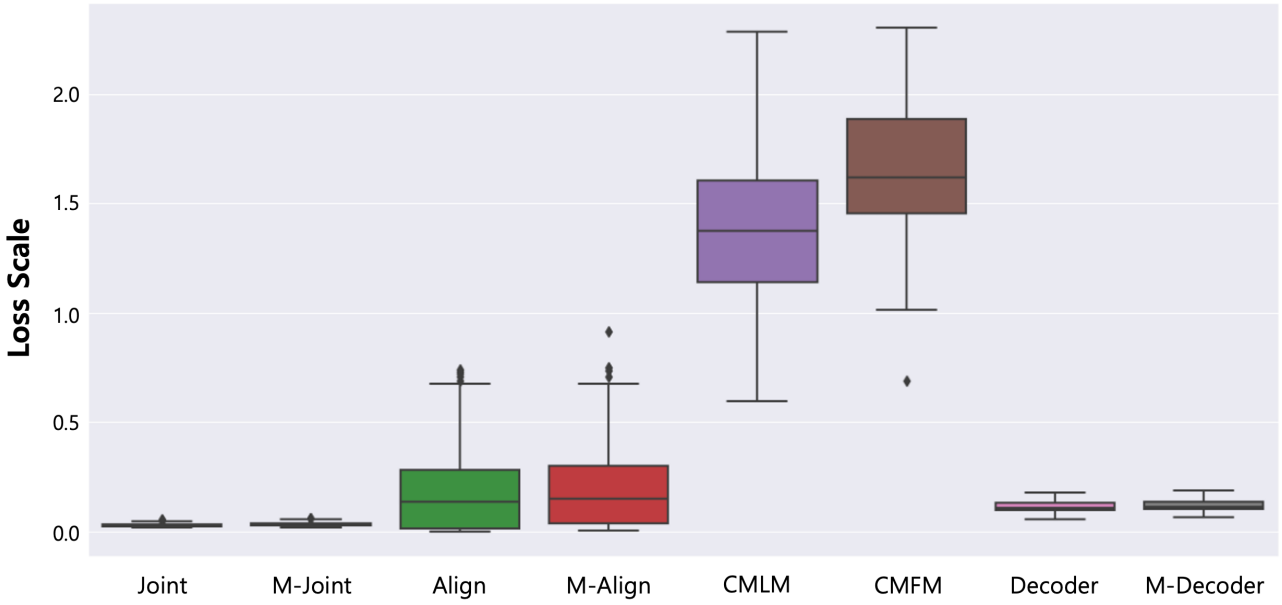

Figure 2. **Loss range for each task.** The ranges of each task loss for each data sample are plotted. A clear distinction is observed between the range of CMLM / CMFM loss and the rest of the task losses.

Table 2. **The effect of MELTR architecture.** Experimental results for different MELTR architectures are provided. The performances are reported for video retrieval on MSRVTT. We do not report performance for task-embedding-only Transformer, as our optimization method is not trained properly in such a setting; $\nabla_w \mathcal{L}^{\text{aux}}$ is always zero.

Architecture	R@1
Linear	27.6
Transformer (SE+TE)	**28.6**
Transformer (SE only)	27.9
Transformer (TE only)	-

E. Effectiveness of input type

In this section, we provide a qualitative analysis for each input type (**SE** only, **TE** only, and **SE + TE**). We visualize $\partial_{\ell_t}\text{MELTR}(\boldsymbol{\ell};\phi)$ denoted in Section 5.2 of the main paper. We calculate it in the same way as in the main paper for three input types on the video captioning task of YouCook2.

Figure 1 illustrates $\partial_{\ell_t}\text{MELTR}(\boldsymbol{\ell};\phi)$ with respect to the scales of the input loss values. When only the **SE** is fed in Figure 1(a), MELTR tends to focus on reasonably challenging samples and downweight the noisy samples as discussed in Section 5.2 of the main paper. Also note that without task information, we observe that the tendency is separated into two clusters with respect to $\partial_{\ell_t}\text{MELTR}(\boldsymbol{\ell};\phi)$: ($\mathcal{L}_{\text{CMLM}}$, $\mathcal{L}_{\text{CMFM}}$) and ($\mathcal{L}_{\text{Joint}}$, $\mathcal{L}_{\text{M-Joint}}$, $\mathcal{L}_{\text{Align}}$, $\mathcal{L}_{\text{M-Align}}$, $\mathcal{L}_{\text{Decoder}}$, $\mathcal{L}_{\text{M-Decoder}}$). We believe that this is because the auxiliary losses are grouped based on the ranges of each loss, as seen in Figure 2, and MELTR distinguishes the tasks to some extent by learning the range of losses without the **TE**. As for the **TE** in Figure 1(b), $\partial_{\ell_t}\text{MELTR}(\boldsymbol{\ell};\phi)$ is obviously invariant to the scale of losses and depend only on the task types. $\mathcal{L}_{\text{Decoder}}$ and $\mathcal{L}_{\text{M-Decoder}}$ rank high because they improve the performance on the video captioning task. In Figure 1(c), MELTR finally takes into account the tasks which are advantageous on the primary task, and guides a learner to focus on a reasonably challenging samples as discussed in Section 5.2 of the main paper, when using the summation of two embeddings (**SE + TE**).

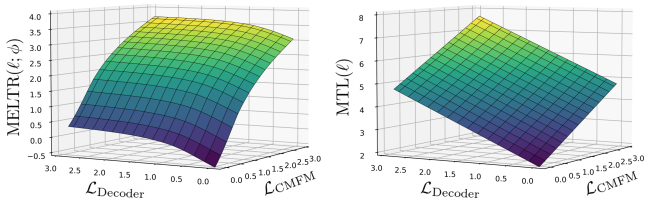

Figure 3. **Non-linearity of MELTR.** MELTR (left) and MTL (right) output with respect to $\mathcal{L}_{\text{Decoder}}$ and $\mathcal{L}_{\text{CMFM}}$.

F. Non-linearity of MELTR

MELTR provides more flexible and effective transformations beyond a simple linear combination of losses through transformer architecture. Table 8 of the main paper evi-

Table 3. **Additional quantitative results.** (Left) The accuracy of video question answering on MSVD-QA is reported. (Middle) The accuracy of action recognition on Kinetics400 is reported. (Right) The accuracy of image classification on CIFAR-100 is reported.

Models	Accuracy
ALPRO	45.9
ALPRO + **MELTR**	**46.8**

Models	Accuracy
Violet	72.4
Violet + **MELTR**	**73.1**

Models	Accuracy
ResNet32	66.5
ResNet32 + **MELTR**	**69.2**

Table 4. **Video captioning on YouCook2.** B3, B4, M, and R mean BLEU-3, BLEU-4, METEOR, and ROUGE-L, respectively. 'Ori.' contains original five auxiliary losses: $\mathcal{L}_{\text{Joint}}$, $\mathcal{L}_{\text{Align}}$, $\mathcal{L}_{\text{CMLM}}$, $\mathcal{L}_{\text{CMFM}}$, and $\mathcal{L}_{\text{Decoder}}$. Also, the last column reports the averaged gain across metrics compared to the Ori. settings of MTL and MELTR, respectively.

Auxiliary losses	Training	B3	B4	M	R	avg. gain
Ori.	MTL	20.68	14.95	20.18	44.25	+0.00
	METLR	**23.47**	**17.29**	**22.25**	**45.67**	+0.00
Ori. + $\mathcal{L}_{\text{M-Decoder}}$	MTL	21.51	15.69	20.73	45.05	+0.73
	MELTR	**23.86**	**17.59**	**22.34**	**46.76**	+0.47
Ori. + $\mathcal{L}_{\text{M-Joint}}$	MTL	21.00	15.19	20.46	44.63	+0.31
	MELTR	**23.76**	**17.53**	**22.22**	**46.63**	+0.37
Ori. + $\mathcal{L}_{\text{M-Align}}$	MTL	20.76	15.01	20.27	44.29	+0.07
	MELTR	**23.55**	**17.45**	**22.16**	**46.56**	+0.26
Ori. + $\mathcal{L}_{\text{M-Decoder}}$ + $\mathcal{L}_{\text{M-Align}}$ + $\mathcal{L}_{\text{M-Joint}}$	MTL	21.72	15.93	20.89	45.16	+0.91
	MELTR	**24.12**	**17.92**	**22.56**	**47.04**	+0.74

dences that MELTR outperforms two linear combinations, the sum of losses (multi-task learning, MTL) and an adaptive and learned linear combination (Meta-Weight Net), by 2.4 and 1.3 R@1 in MSRVTT for text-to-video retrieval. Qualitatively, Figure 3 shows the non-linearity of MELTR in contrast to the multi-task learning (MTL) by visualizing their outputs given two input losses: $\mathcal{L}_{\text{Decoder}}$ and $\mathcal{L}_{\text{CMFM}}$.

G. Effectiveness of advanced loss of UniVL

For video captioning on YouCook2, in order of importance, the losses can be sorted as $\mathcal{L}_{\text{M-Decoder}}$, $\mathcal{L}_{\text{M-Joint}}$, and $\mathcal{L}_{\text{M-Align}}$. Table 4 shows the additional ablation study on newly added losses. First, using all three newly added losses improves the performance with both MTL (+0.91) and MELTR (+0.74) on average. As for the individual loss, by adding $\mathcal{L}_{\text{M-Decoder}}$, the average performance gain of MELTR is 0.47. On the other hand, with $\mathcal{L}_{\text{M-Joint}}$ or $\mathcal{L}_{\text{M-Align}}$, the performance gap is decreased to 0.37 and 0.26 respectively, implying that they are relatively less effective for video captioning than $\mathcal{L}_{\text{M-Decoder}}$ as observed in Sec. 5.2 of the main paper.

H. Adaptation to a new baseline and tasks

Plug-in to a new baseline. In Table 3 (Left), we conduct an experiment with another strong model ALPRO [28] trained with four pretext losses. In the video question answering task on MSVD-QA, ALPRO shows the original performance of 45.9%, and MELTR improves it to 46.8%.

Video only setting. We also evaluate action recognition performance on Kinetics400 [29] by applying MELTR to Violet in Table 3 (Middle). Since the action recognition is a unimodal task with *only* 'videos', we use the following two losses: classification loss (primary task) and Masked Visual-token Modeling loss (MVM; auxiliary task). Violet's accuracy is improved from 72.4% to 73.1%.

Image only setting. Furthermore, to verify the generalizability of MELTR to other domains, we also conduct our experiment on the 'image' domain (image classification on CIFAR-100) with ResNet32 backbone in Table 3 (Right). We add two simple auxiliary losses (mixup [30] and rotation [31]) with a basic classification loss. Our MELTR outperforms the baseline by a margin of 2.7%. These experimental results demonstrate that MELTR is a general framework to be adapted to a wide range of domains and tasks.

References

[1] Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. Univl: A unified video and language pre-training model for multimodal understanding and generation, 2020. 1

[2] Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. https://github.com/microsoft/UniVL, 2020. 1

[3] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, 2019. 1

[4] Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. Violet: End-to-end video-language transformers with masked visual-token modeling. *arXiv preprint arXiv:2111.12681*, 2021. 1

[5] Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. https://github.com/tsujuifu/pytorch_violet, 2021. 1

[6] Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. Merlot: Multimodal neural script knowledge models. In *NeurIPS*, 2021. 1

[7] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021. 1

[8] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*, 2018. 1

[9] Alex Jinpeng Wang, Yixiao Ge, Rui Yan, Yuying Ge, Xudong Lin, Guanyu Cai, Jianping Wu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. All in one: Exploring unified video-language pre-training. *arXiv preprint arXiv:2203.07303*, 2022. 1

[10] Alex Jinpeng Wang, Yixiao Ge, Rui Yan, Yuying Ge, Xudong Lin, Guanyu Cai, Jianping Wu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. https://github.com/showlab/all-in-one, 2022. 1

[11] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021. 1

[12] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1

[13] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 2017. 1

[14] Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. In *NeurIPS*, 2011. 1

[15] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002. 1

[16] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *ACLW*, 2005. 1

[17] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. *Text summarization branches out*, 2004. 1

[18] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, 2015. 1

[19] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 1

[20] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, 2018. 1

[21] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, 2016. 1

[22] Youngjae Yu, Jongseok Kim, and Gunhee Kim. A joint sequence fusion model for video question answering and retrieval. In *ECCV*, 2018. 1

[23] Antoine Miech, Ivan Laptev, and Josef Sivic. Learning a text-video embedding from incomplete and heterogeneous data. *arXiv preprint arXiv:1804.02516*, 2018. 1

[24] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *CVPR*, 2017. 2

[25] Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. Video question answering via gradually refined attention over appearance and motion. In *MM*, 2017. 2

[26] David Chen and William B Dolan. Collecting highly parallel data for paraphrase evaluation. In *ACL*, 2011. 2

[27] Amir Zadeh, Rowan Zellers, Eli Pincus, and Louis-Philippe Morency. Mosi: multimodal corpus of sentiment intensity and subjectivity analysis in online opinion videos. *arXiv preprint arXiv:1606.06259*, 2016. 2

[28] Dongxu Li, Junnan Li, Hongdong Li, Juan Carlos Niebles, and Steven CH Hoi. Align and prompt: Video-and-language pre-training with entity prompts. In *CVPR*, 2022. 4

[29] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017. 4

[30] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018. 4

[31] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 4